Approximate TSP in Graphs with Forbidden Minors

Michelangelo Grigni*

Dept. of Mathematics and Computer Science,
Emory University, Atlanta Georgia 30322, USA,
mic@mathcs.emory.edu

Keywords: graph minor, genus, separator, spanner, TSP, approximation scheme.

Abstract. Given as input an edge-weighted graph, we analyze two algorithms for finding subgraphs with low total edge weight. The first algorithm finds a separator subgraph with a small number of components, and is analyzed for graphs with an arbitrary excluded minor. The second algorithm finds a spanner with small stretch factor, and is analyzed for graphs in a hereditary family $\mathcal{G}(k)$. These results imply (i) a QPTAS (quasi-polynomial time approximation scheme) for the TSP (traveling salesperson problem) in unweighted graphs with an excluded minor, and (ii) a QPTAS for the TSP in weighted graphs with bounded genus.

1 Introduction

In the *traveling salesperson problem* (TSP) we are given n sites and their distance matrix, and our goal is to find a simple closed tour of the sites with minimum total distance. The TSP has driven both practical and theoretical algorithm research for several decades [9]. Most variants are NP-hard, and therefore much attention is given to approximate solutions for *metric TSP*, where the distance matrix is a metric (nonnegative, symmetric, and satisfying the triangle inequality). An algorithm of Christofides [6] finds a metric TSP solution with cost at most 3/2 times optimal. We would prefer a polynomial time approximation scheme (PTAS); that is, for each $\varepsilon > 0$, we would like a polynomial time algorithm which produces a solution with cost at most $1+\varepsilon$ times optimal. However, metric TSP is MAXSNP-hard even when all distances are one or two [12], and so there is some positive ε_0 such that finding a $1+\varepsilon_0$ approximation is NP-hard. Indeed, the 3/2 guarantee of Christofides has not been improved (although in practice, other heuristics are much better).

However, there has been recent progress in certain restricted metric spaces. In [8] we found a PTAS for the TSP on the nodes of an unweighted planar graph, where the metric is given by shortest path lengths in the graph. We later generalized [4] this to allow distances defined by non-negative edge costs. Arora [3] (also Mitchell [11]) gave a PTAS for the TSP and related problems

* Supported by NSF Grant number CCR-9820931.

for points in a Euclidean space of fixed dimension. Roughly speaking, all of these results depend on the ability to find inexpensive and "well-connected" separators.

In this paper we extend the methods of [4] from planar graphs to larger graph families. This leads us to a general notion of well-connected separator that may have other algorithmic applications. We present two subgraph finding algorithms; in both algorithms the goal is a *light* subgraph, meaning that it has low total edge weight. In Section 3 we give an algorithm finding a light separator subgraph with few connected components, in graphs from any family with a nontrivial excluded minor. In Section 4 we give an algorithm finding a light spanner with low stretch factor, in graphs from a family $\mathcal{G}(k)$, to be defined. This family includes graphs of bounded genus.

Finally, in Section 5 we sketch a QPTAS (quasi-polynomial time approximation scheme) for metric TSP in two situations. First, for the shortest path metric in an unweighted graph from a family with an excluded minor. Second, for the shortest path metric in an edge-weighted graph from family $\mathcal{G}(k)$, for any fixed k. Both schemes run in time $n^{O(\log\log n)}$.

2 Preliminaries

All graphs in this paper are undirected and simple. A graph $G = (V, E)$ is *edge-weighted* if it has a non-negative weight (or length) $\ell(e)$ on each edge $e \in E$; it is *vertex-weighted* if it has a non-negative weight $w(v)$ on each vertex $v \in V$. A subgraph H of G inherits these weights on its edges and vertices. The total edge weight and vertex weight in H are denoted by $\ell(H)$ and $w(H)$, respectively. The number of connected components in H is denoted $\#(H)$.

When G is edge-weighted, $d_G(u, v)$ denotes the minimum length $\ell(P)$ of a path P connecting endpoints u and v. This is zero when $u = v$, and $+\infty$ when u and v are disconnected. $G' = (V, E')$ is a *spanning subgraph* if it spans each component of G. Clearly $d_G(u, v) \leq d_{G'}(u, v)$; the *stretch factor* $\mathrm{sf}(G', G)$ is the minimum s such that $d_{G'}(u, v) \leq s \cdot d_G(u, v)$ for all $u, v \in V$ (it suffices to consider only edge pairs $\{u, v\} \in E$). When $\mathrm{sf}(G', G) \leq s$, we say that G' is an *s-spanner* in G.

Given a graph G, a *minor* of G is a graph resulting from some sequence of edge deletion, vertex deletion, or edge contraction operations (denoted $G - e$, $G - v$, and G/e respectively). Since we only consider simple graphs, we discard self-loops and parallel edges. We say G has an H-minor (denoted $H < G$) if G has a minor isomorphic to H. A *hereditary graph property* is a class $\mathcal{P}$ of graphs closed under isomorphism, such that whenever G is in $\mathcal{P}$, so are its minors. In a series of papers, Robertson and Seymour show that every hereditary graph property is characterized by a finite set of forbidden minors (see [7] for an overview). The prime example is Kuratowski's characterization of planar graphs by the forbidden minors $\{K_5, K_{3,3}\}$.

For a subset X of vertices in G, an X-flap is the vertex set of a connected component of $G - X$. Given a vertex-weighted graph G, a *separator* is a subgraph

S such that every $V(S)$-flap has weight at most $w(G)/2$. Note that separators are usually defined as just a vertex set, but in this paper we are interested in a tradeoff between $\ell(S)$ and $\#(S)$.

Let $V(G)^{\leq k}$ denote the collection of sets of at most k vertices in G. A *haven of order* k is a function β assigning an X-flap to each $X \in V(G)^{\leq k}$, such that $\beta(Y) \subseteq \beta(X)$ whenever $X \subseteq Y \in V(G)^{\leq k}$. Given a vertex-weighted graph G and a non-separator vertex subset X, let $\beta^w(X)$ denote the unique X-flap with weight exceeding $w(G)/2$. If X is a separator, let $\beta^w(X) = \emptyset$. Note that if G has no separator of size k, then β^w (restricted to $V(G)^{\leq k}$) is a haven of order k.

3 A Well-Connected Separator

Alon, Seymour, and Thomas [1] give a polynomial time algorithm to find a separator in a graph with an excluded minor. Specifically, given as input a vertex-weighted graph G and a graph H, their algorithm either finds an H-minor in G, or it finds a separator in G with at most $h^{3/2}n^{1/2}$ vertices, where h is the number of vertices in H. In particular, if we fix a non-trivial hereditary graph property $\mathcal{P}$ and only consider inputs $G \in \mathcal{P}$, then this algorithm finds separators of size $O(n^{1/2})$; this generalizes the planar separator theorem [10].

They (and we) only consider the case $H = K_h$, since a K_h-minor implies an H-minor. A *covey* is a forest $\mathcal{C}$ in G such that each pair of component trees is connected by an edge of G. A covey with $\#(\mathcal{C}) = h$ witnesses a K_h-minor.

In our application, G is also edge-weighted. We modify their algorithm to allow a trade-off between the total edge weight of the separator and the number of its connected components. We claim the following:

Theorem 1. *There is a polynomial time algorithm taking as input a vertex-weighted edge-weighted graph G, a positive integer h, and a positive real $\varepsilon \leq 1$, and which produces as output either:*
(a) a K_h-minor in G, or
(b) a separator S of G such that $\ell(S) \leq \varepsilon\, h\, \ell(G)$ and $\#(S) \leq h^2/\varepsilon$.

We use much of their algorithm unchanged, so in this abstract we simply describe and analyze our changes. Our basic subroutine is the following slight modification of [1, Lemma 2.1]:

Lemma 2. *Let G be an edge-weighted graph with m edges, let $A_1, \ldots, A_k$ be subsets of $V(G)$, and let ε be a positive real number. There is a polynomial time algorithm which returns either:*
(i) a tree T in G such that $\ell(T) \leq \varepsilon\, \ell(G)$ and $V(T) \cap A_i \neq \emptyset$ for $i = 1, \ldots, k$; or
(ii) a set $Z \subseteq V(G)$ such that $|Z| \leq (k-1)/\varepsilon$ and no Z-flap intersects all of $A_1, \ldots, A_k$.

The proof of this lemma is essentially the same, except that we use a shortest paths tree rather than breadth first search. The rest of the proof is omitted.

The algorithm is iterative. After t steps of the algorithm, we have a subgraph X_t and a covey $\mathcal{C}_t$; initially X_0 and $\mathcal{C}_0$ are empty. In step t of the algorithm, we halt if either $\#(\mathcal{C}_{t-1}) \geq h$ or X_{t-1} is a separator. Otherwise, we let

$B_{t-1} = \beta^w(X_{t-1})$ and we invoke Lemma 2 on $G[B_{t-1}]$, where the A_i's are the neighborhoods of the component trees in $\mathcal{C}_{t-1}$; we call this step either a T-step or a Z-step, depending on which is returned. The returned T or Z is then used to define X_t and $\mathcal{C}_t$, according to several cases as described in [1]. We have these invariants:

1. X_t is a subgraph of $\mathcal{C}_t$.
2. For each component tree C in $\mathcal{C}_t$, either $X_t \cap C$ equals some T returned in a T-step, or $X_t \cap C$ is a set of disconnected vertices contained in some Z returned in a Z-step.
3. $B_t \subseteq B_{t-1}$; and if these are equal, then $X_t \subset X_{t-1}$.
4. $V(\mathcal{C}_t)$ and B_t are disjoint.

By the first invariant, X_t is the union of at most h parts of the form $X_t \cap C$. By the second invariant and Lemma 2, each part has $\ell(X_t \cap C) \leq \varepsilon \cdot \ell(G)$ and $\#(X_t \cap C) \leq (k-1)/\varepsilon$. Therefore $\ell(X_t) \leq h\ \varepsilon\ \ell(G)$ and $\#(X_t) \leq h^2/\varepsilon$, as required in Theorem 1(b).

By invariant 3 above, we see that the sequence of pairs $(|B_t|, |X_t|)$ is lexicographically decreasing; therefore the algorithm halts after at most n^2 iterations. In fact an improved time analysis is possible, but we omit it here.

Remark. Our algorithm (and the original) may also be useful in situations where we have a G with no K_h-minor, but the vertex-weighting w is unknown. Observe that w affects the algorithm in only two ways. First, it can tell us when to stop because X_t is a separator. Second, when we update X_{t-1}, B_{t-1} splits into disjoint flaps, and w tells us which flap to take as the next B_t. Since the B_t's decrease, the tree of possible computations (depending on w) has at most n leaves. The tree depth is at most the maximum number of iterations, considered above. Therefore there is a polynomial size collection of vertex sets in G, such that for any weighing w, one of them is a separator satisfying the conditions of Theorem 1(b).

4 The Span Algorithm

Althöfer *et al.* [2] introduced the following greedy algorithm to find an s-spanner in an edge-weighted graph G. The parameter s is at least one, and $d_{G'}(e)$ denotes the length of the shortest path in G' connecting the endpoints of edge e (the length may be $+\infty$):

```
Span(G = (V, E), s):
    G' ← (V, ∅)
    for all e ∈ E in non-decreasing ℓ order do
        if s · ℓ(e) < d_G'(e) then add e to G'
    return G'
```

In the resulting G', we have $d_{G'}(e) \leq s \cdot \ell(e)$ for every edge $e \in E$; therefore G' is an s-spanner. By comparison with Kruskal's algorithm, we see that $T(G) \stackrel{\text{def}}{=}$

$\mathrm{Span}(G, n-1)$ is a minimum spanning forest in G, and $\mathrm{Span}(G, s)$ always contains $T(G)$. The Span algorithm is idempotent in this sense: if $G' = \mathrm{Span}(G, s)$, then $G' = \mathrm{Span}(G', s)$.

efine the *tree weight* of G' as $\mathrm{tw}(G') = \ell(G')/\ell(T(G'))$ (note that $T(G') = T(G)$). We seek a tradeoff between $\mathrm{sf}(G', G)$ and $\mathrm{tw}(G')$. The algorithm has two extremes: $\mathrm{Span}(G, n-1)$ has tree weight one but may have stretch factor $n-1$; $\mathrm{Span}(G, 1)$ has stretch factor one but may have tree weight nearly $n^2/4$. For intermediate s, the following tradeoff is known [2, Thm. 2]:

Theorem 3. *If $s > 1$ and G is planar then* $\mathrm{tw}(\mathrm{Span}(G, s)) \leq 1 + 2/(s-1)$.

With s close to one, this theorem is a critical element of the approximation scheme for the TSP in weighted planar graphs [5]. Motivated by this application, we seek to extend the result to larger graph families.

Definition 4. *Suppose G is a graph, ℓ is an edge weighting in G, and T is a spanning forest. Define:*

1. $\mathrm{gap}_\ell(e) = d_{G-e}(e) - \ell(e)$.
2. $\mathrm{gap}(G, T) = \max_\ell(\sum_{e \notin T} \mathrm{gap}_\ell(e))/\ell(T)$, *where ℓ ranges over all edge weightings such that $T = T(G)$.*
3. $\mathrm{gap}(G) = \max_T \mathrm{gap}(G, T)$, *where T ranges over all spanning forests.*
4. *the graph class* $\mathcal{G}(k) = \{G \mid \mathrm{gap}(G) \leq k\}$.

Remark. Given G and T, $\mathrm{gap}(G, T)$ is the value of a linear program; suppose ℓ achieves the maximum. If e is a cut edge then $\mathrm{gap}_\ell(e)$ is infinite, but it does not matter since $e \in T$ (and we may set $\ell(e) = 0$). For other edges e we may assume $\mathrm{gap}_\ell(e) \geq 0$, because otherwise we could improve ℓ by setting $\ell(e) = d_{G-e}(e)$.

Theorem 5. *$\mathcal{G}(k)$ is a hereditary graph property.*

Proof. $\mathcal{G}(k)$ is easily closed under isomorphism; we need to show $\mathrm{gap}(H) \leq \mathrm{gap}(G)$ whenever $H < G$. Take ℓ and T' in H such that $\mathrm{gap}(H, T') = \mathrm{gap}(H)$. In G we will define an edge weighting (also denoted ℓ) and a spanning forest T. By induction it suffices to consider these three cases:
Case $H = G - e$: If e connects two components of $G - e$, let $\ell(e) = 0$ and include e in T so $T - e = T'$. Otherwise let $\ell(e) = d_{G-e}(e)$ and $T = T'$.
Case $H = G - v$: By deleting edges first, we may assume v is isolated. Then ℓ is unchanged, and we add the isolated v to T.
Case $H = G/e$: By deleting edges first, we may assume that no two edges in G merge to one in G/e. Let $\ell(e) = 0$ and include e in T so that $T/e = T'$.

In all cases we have constructed ℓ and T such that $(\sum_{e \notin T} \mathrm{gap}_\ell(e))/\ell(T) = \mathrm{gap}(H, T')$, therefore $\mathrm{gap}(G) \geq \mathrm{gap}(H)$ as required. ☐

Let $g(H)$ denote the girth of a graph H. By considering the uniform edge weighting $\ell \equiv 1$, we have:

Corollary 6. $\mathrm{gap}(G) \geq \max_H (g(H) - 2)(|E(H)|/(|V(H)| - 1) - 1)$, *where H ranges over all 2-connected minors of G.*

We now relate $\mathrm{gap}(G)$ to the Span algorithm. Let $\mathrm{gap}'_\ell(e)$ denote the edge gap in G', that is $d_{G'-e}(e) - \ell(e)$.

Lemma 7. *If* $G' = \mathrm{Span}(G, s)$*, then* $\ell(e) < 1/(s-1) \cdot \mathrm{gap}'_\ell(e)$ *for all* e *in* G'.

Proof. We follow [2, Lem. 3]. Let P be the shortest path in $G' - e$ connecting the endpoints of e. Just before the algorithm inserts the longest edge $f \in \{e\} \cup P$, we have $s \cdot \ell(e) \le s \cdot \ell(f) < d_{G'-f}(f) \le \ell(P) = d_{G'-e}(e)$. □

Theorem 8. *If* $s > 1$ *and* $G \in \mathcal{G}(k)$*, then* $\mathrm{tw}(\mathrm{Span}(G, s)) \le 1 + k/(s-1)$.

Proof. We are given G with some weighting ℓ. Let $G' = \mathrm{Span}(G, s)$; we need to show $\mathrm{tw}(G') \le 1 + k/(s-1)$. Theorem 5 implies $\mathrm{gap}(G') \le k$. Let $T = T(G')$. By the definition of $\mathrm{gap}(G')$, $\sum_{e \notin T} \mathrm{gap}'_\ell(e) \le k\ell(T)$. By the lemma we have $\sum_{e \notin T} \ell(e) \le k/(s-1)\ell(T)$, and the result now follows. □

Remark. Theorem 3 is proved by showing $\mathcal{G}(2)$ contains all planar graphs. Although not stated in this way, they construct a feasible point in a linear program dual to the definition of $\mathrm{gap}(G, T)$ (it is feasible even if we drop the $T = T(G)$ constraint).

Lemma 6 implies that K_h is a forbidden minor in $\mathcal{G}(h/2 - 1 - \varepsilon)$; we conjecture a converse relation.

Conjecture 9. There is a function $f(\cdot)$ such that $\mathcal{G}(f(h))$ contains all graphs with no K_h-minor.

Absent this conjecture, we offer weaker evidence that $\mathcal{G}(k)$ is interesting:

Lemma 10. *Suppose* G *has genus* g*; that is,* G *may be drawn without crossings on an orientable surface with* g *handles. Then* $G \in \mathcal{G}(12g - 4)$.

Proof. (Sketch.) Suppose G is drawn in an orientable surface with g handles. Choose a spanning tree T such that $\mathrm{gap}(G, T) = \mathrm{gap}(G)$. For edges $e, f \notin T$, say they are equivalent if the cycles in $e + T$ and $f + T$ are homotopic. If we take T and all the edges of one equivalence class, we get a planar subgraph of G. Suppose there are h equivalence classes; then G is the union of h planar subgraphs $G_1, \ldots, G_h$ with a common spanning tree T. By Definition 4 we see $\mathrm{gap}(G, T) \le \sum_{i=1}^{h} \mathrm{gap}(G_i, T)$, and this is at most $2h$.

It now suffices to show $h \le 6g - 2$. We contract T to a point p, so the arcs become non-crossing non-homotopic loops based at p. We pick one loop of each class, and consider the faces defined by these h loops. We may assume that each face is a 2-cell, otherwise we could add another loop. Since no two loops are homotopic, no face has two sides. There may be one face bounded by one loop e, but then the other side of e has at least four sides; all other faces have at least three sides. Therefore $2e = \sum_{\Delta} |\Delta| \ge 3f - 1$, where e is the number of loops, f is the number of faces, and $|\Delta|$ is the number of sides of face Δ. Combining this with Euler's formula $v - e + f = 2 - 2g$ gives our bound (here $v = 1$). A simple construction shows $h = 6g - 2$ is achievable for $g \ge 1$. □

5 The Approximation Schemes

We will reuse the methods introduced for planar graphs [8, 4], and only sketch them here. We are given as input a connected graph G, and a parameter $\varepsilon > 0$. Our goal is to find a circuit in the graph visiting each vertex at least once, and with length within $1 + \varepsilon$ times the minimum (this is equivalent to the original metric TSP formulation). The minimum lies between $\ell(T(G))$ and $2\ell(T(G))$, so it suffices to find a solution with additive error at most $\varepsilon\ell(T(G))$. We need to handle these two cases:

Case G is unweighted and has no K_h-minor: We introduce the uniform edge weighting $\ell \equiv 1$. By Mader's Theorem [7, 8.1.1] there is a constant K such that $\ell(G) \leq K\ell(T(G))$ (the best K is $\Theta(h\sqrt{\log h})$ [14]).

Case G is weighted and in $\mathcal{G}(k)$: We replace G by $\mathrm{Span}(G, 1 + \varepsilon/4)$; this substitution introduces at most $(\varepsilon/2)\ell(T(G))$ additive error. Theorem 8 implies $\ell(G) \leq K\ell(T(G))$, where $K = 1 + 4k/\varepsilon$. Also by Lemma 6 we know G contains no K_h-minor, for $h \geq 2(k + 1)$.

Now in either case we know that G has no K_h-minor, and that $\ell(G) \leq K\ell(T)$. We now need to find a circuit within $(\varepsilon/2)\ell(T(G))$ of optimal in time $n^{O(\log\log n)}$, where the hidden constant depends on ε, h, and K.

Given a separator S of G, it is easy to find a *separation*: that is a triple (S, A_1, A_2) such that S is a subgraph, $A_1 \cup A_2 = V(G)$, $A_1 \cap A_2 = V(S)$, there are no edges between $A_1 - S$ and $A_2 - S$, and each $A_i - S$ has vertex weight at most $(2/3)w(G)$. So by Theorem 1, we have:

Corollary 11. *Suppose G is an edge-weighted graph with no K_h-minor, and $\delta \leq 1$ is a positive real number. Then there is a polynomial time algorithm finding a separation (S, A_1, A_2) of G such that $\ell(S) \leq \delta\,\ell(G)$ and $\#(S) \leq h^3/\delta$.*

We give G a uniform vertex-weighting w. We will build a linear size decomposition tree $\mathcal{T}$ of G, by repeated application of Corollary 11 with the parameter $\delta = \gamma\varepsilon/\log n$, where $\gamma > 0$ is a constant to be determined.

If a weighted graph F in $\mathcal{T}$ has less than $\Theta(\delta^{-2})$ vertices, then it is a leaf. Otherwise, we apply Corollary 11 to find a separation (S_F, A_1, A_2) in F. For each A_i we let F_i denote the graph that results from $F[A_i]$ by contracting each component of S_F to a point; F_1 and F_2 are the children of F in $\mathcal{T}$. We call the new contracted points *portal points*, and give them (for now) zero weight. Note that F_i may also inherit portal points from F, and that each edge of F appears in at most one child.

Since $w(F_i) \leq 2/3w(F)$, the depth of $\mathcal{T}$ is $O(\log n)$. We introduce at most $f = h^3/\delta$ new portals in each split, so every graph in $\mathcal{T}$ has at most p portals, where $p = O(f \log n) = O(h^3(\log^2 n)/\varepsilon)$. Since each original edge of G appears at most once in a level of $\mathcal{T}$, the edges of all S_F contracted in a single level have total weight at most $\delta\,\ell(G)$. Summing over all levels, the total weight of all the contracted spanners is $O(\varepsilon\ell(G))$. By a suitable choice of $\gamma = \Theta(1/K)$, we may ensure that this is at most $(\varepsilon/4)\ell(T(G))$.

Consider the optimum circuit τ in G. After performing the splits and contractions, τ has an image τ_F in each graph F of $\mathcal{T}$; τ_F enters and leaves F through its portals in some order, defining a sequence of portal-terminated paths within F, covering its vertices. Furthermore, by a simple patching argument [4, Lemma 3.2], we may rearrange τ (without changing its cost) so that each τ_F uses each portal of F at most twice as an endpoint.

Therefore, we are led to the following problem, which we will solve approximately by dynamic programming in the tree $\mathcal{T}$. Given a graph $F \in \mathcal{T}$ and an sequence σ of its portals where each portal appears at most twice in σ, a σ-tour of F is a sequence of paths covering F, with path endpoints as specified by the list σ. For each σ, we want to find a near-optimal σ-tour in F.

If F is a leaf in $\mathcal{T}$, then we exactly solve each such problem in $2^{O(1/\delta)}$ time, using the ordinary minor separator theorem [1]. If F has children, then after we have solved all such problems for F_1 and F_2, we may solve them for F as follows. Consider all pairings of a σ_1-tour in F_1 and a σ_2-tour in F_2; if they are compatible, their paths patch together to give us some σ-tour in F. For each σ, we record the cheapest combination obtained; we then recover a true σ-tour in F by "uncontracting" the edges of S_F and charging each uncontracted edge at most twice.

As shown above, the total weight of all these charged edges (over all of $\mathcal{T}$) is at most $(\varepsilon/4)\ell(T(G))$, therefore the total additive error in these contributed by this uncontraction is $(\varepsilon/2)\ell(T(G))$. We can show that this is the only source of additive error in our solution, so we have the promised approximation scheme.

The time of the above algorithm is roughly the number of dynamic programming subproblems, which is $n^{O(1)}p^{O(p)}$. With our previous bound for p, this is $n^{O((Kh^3/\varepsilon)\log n \log\log n)}$. In fact we can do better; by using the portal weighing scheme of [8], we can ensure that each graph in $\mathcal{T}$ has at most $p = 6f$ portals, while $\mathcal{T}$ still has $O(\log n)$ depth. With this improvement, our time bound is $n^{O((Kh^3/\varepsilon)\log\log n)}$.

6 Open Problems

Of course proving Conjecture 9 would help unify our present results.

We would prefer a true polynomial time approximation scheme, rather than quasi-polynomial. Our obstacle is the number of portal orderings σ that we must consider for each F. In the case of planar graphs [4], we overcame the obstacle by observing that we only needed to consider those σ-tours corresponding to non-crossing paths in an embedding of F on a sphere with $O(1)$ holes. This observation reduces the number of σ considered to a simple exponential $2^{O(p)}$, and consequently the total time is polynomial in n. In the present situation, a bound like the above is unknown, but it is at least plausible. This is because graphs with a forbidden minor are characterized [13] in terms of blocks that can be "nearly drawn" on a 2-manifold with a bounded genus and number of holes.

We would also like to solve the Steiner version of the problem, where along with G we are given a set of "terminal" vertices, and we want to find a minimum length tour visiting all the terminals. The remark at the end of Section 3 is a preliminary step in that direction.

Acknowledgment: We thank Robin Thomas for help with Lemma 10.

References

1. N. Alon, P. D. Seymour, and R. Thomas. A separator theorem for graphs with an excluded minor and its applications. In *Proc. 22nd Symp. Theory of Computing*, pages 293–299. Assoc. Comput. Mach., 1990.
2. I. Althöfer, G. Das, D. P. Dobkin, D. Joseph, and J. Soares. On sparse spanners of weighted graphs. *Discrete Comput. Geom.*, 9(1):81–100, 1993. An early version appeared in SWAT'90, LNCS V. 447.
3. S. Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *Journal of the ACM*, 45(5):753–782, Sept. 1998.
4. S. Arora, M. Grigni, D. Karger, P. Klein, and A. Woloszyn. A polynomial-time approximation scheme for weighted planar graph TSP. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 33–41, San Francisco, California, 25–27 Jan. 1998.
5. S. Arora, M. Grigni, D. Karger, P. Klein, and A. Woloszyn. A polynomial-time approximation scheme for weighted planar graph TSP. In *9th Annual ACM-SIAM Symp. on Discrete Algorithms*, pages 33–41, Jan. 1998.
6. N. Christofides. Worst-case analysis of a new heuristic for the traveling salesman problem. In J. F. Traub, editor, *Symposium on New Directions and Recent Results in Algorithms and Complexity*, page 441, NY, 1976. Academic Press. Also CMU Tech. Report CS-93-13, 1976.
7. R. Diestel. *Graph theory.* Springer-Verlag, New York, 1997. Translated from the 1996 German original.
8. M. Grigni, E. Koutsoupias, and C. Papadimitriou. An approximation scheme for planar graph TSP. In *36th Annual Symposium on Foundations of Computer Science (FOCS'95)*, pages 640–645, Los Alamitos, Oct. 1995. IEEE Computer Society Press.
9. E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys. *The Traveling Salesman Problem.* Wiley, 1985, 1992.
10. R. J. Lipton and R. E. Tarjan. A separator theorem for planar graphs. *SIAM Journal on Applied Mathematics*, 36:177–189, 1979.
11. Mitchell. Guillotine subdivisions approximate polygonal subdivisions: A simple polynomial-time approximation scheme for geometric TSP, k-MST, and related problems. *SICOMP: SIAM Journal on Computing*, 28, 1999.
12. C. H. Papadimitriou and M. Yannakakis. The traveling salesman problem with distances one and two. *Mathematics of Operations Research*, 18:1–11, 1993.
13. N. Robertson and P. D. Seymour. Graph minors XVII: Excluding a non-planar graph. Submitted.
14. A. Thomason. An extremal function for contractions of graphs. *Math. Proc. Cambridge Philos. Soc.*, 95(2):261–265, 1984.